FlightFinder: Navigating Your Air Travel Options

Team Members:\ Smily Nakka (Team Lead)\ Dharanikota Vimala Subhashini\ Sangidi Mohan Lakshman\ Shruti Singh

Faculty Mentor: Anji Babu\ **Industry Mentor:** No mentor assigned

---

Project Introduction

The FlightFinder application is a comprehensive digital platform built to enhance the airline ticket booking experience. Designed with modern users in mind, this project leverages full-stack web development technologies to create an interactive, reliable, and secure application. Whether you're a frequent flyer or an occasional traveler, this application streamlines every step from flight search to ticket confirmation.

Key features include an intuitive user interface, advanced filtering, personalized bookings, robust payment systems, and secure authentication for both users and admins.

Scenario Use Case

John, a frequent business traveler, needs to book a flight to Paris for a conference. He uses FlightFinder to:

- Enter travel details (New York → Paris, April 10–15, Business class)
- View direct flights and apply filters
- Select his preferred airline
- Pick a window seat with extra legroom
- Make secure payment
- Receive an instant e-ticket and confirmation

This scenario demonstrates how the application simplifies complex travel planning into a seamless process.

Technical Architecture

The FlightFinder system consists of three main components:

- **Frontend**: Built using React.js to deliver a responsive and user-friendly experience. Handles search, booking, login, and dashboards.
- **Backend**: Implemented using Node.js and Express.js. Manages API requests, authentication, and business logic.
- **Database**: MongoDB is used to store user profiles, flight data, and booking history.

An ER diagram outlines key relationships among users, bookings, payments, and flights.

Pre-Requisites for Development

- Node.js and npm (Download)
- MongoDB (Download)
- Express.js (`npm install express`)
- React.js (Create App Guide)
- HTML, CSS, JavaScript basics
- Mongoose for MongoDB interaction
- Axios for API requests
- Git (Download Git)
- Visual Studio Code (Download)

To connect MongoDB to Node.js: Mongoose Guide

Project Repository and Setup

GitHub Repository: FlySmily - BookSmarter TravelLighter

Steps to run:

```
# Clone the repository
https://github.com/Smily2531/FlySmily-BookSmarter-TravelLighter.git

# Navigate and install frontend dependencies
cd client
npm install

# Navigate and install backend dependencies
cd ../server
npm install

# Run servers
npm run dev   # for backend
npm start     # for frontend
```

Visit: `http://localhost:3000` to access the application.

Project Structure

Client (Frontend):

- Built with ReactJS
- Uses Axios, Bootstrap
- Manages search, booking, login, dashboard

Server (Backend):

- Built with Node.js + Express.js
- Uses MongoDB via Mongoose
- API routes for Users, Flights, Admin, Bookings

Backend Development

- Create MongoDB collections for flights, users, and bookings
- Implement APIs for authentication and CRUD operations
- Secure admin routes and user bookings
- Use middleware for error handling and validation

Frontend Development

- User registration and login components
- Flight search filters by location, date, class
- Bookings with seat selection
- Admin panel to:
- Add/edit/delete flights
- View user activity and bookings

Application Flow

User Journey:

- Register/Login
- Search flights
- Book flights and choose seats
- Make secure payments
- View/cancel bookings

Admin Journey:

- Secure login
- Manage all flights, bookings, and users

Project Timeline

- Week 1: Requirement Analysis and Stack Setup
- Week 2–3: Backend Development
- Week 4–5: Frontend Interface
- Week 6: Integration and Testing
- Week 7: Bug Fixes and Optimization

- Week 8: Deployment and Documentation

References & Demonstrations

- Backend Code Walkthrough
- Frontend Demo
- Live Application Demo

Conclusion

The "FlightFinder" project marks a significant milestone in the practical application of web development skills. It has allowed our team to deeply engage with full-stack technologies, ranging from React.js and Node.js to MongoDB and Express. This project not only enhanced our technical skills but also improved our ability to work collaboratively in a team environment. From planning and coding to debugging and documentation, every phase of this project has been a valuable learning experience. With its real-world utility and user-focused design, FlightFinder stands as a testament to what students can achieve with the right tools, guidance, and teamwork.

Prepared by: Smily Nakka\ **B.Tech, 3rd Year, CSE**\ **Aditya College of Engineering & Technology (A)**